IN THE CLAIMS

What is claimed is:

1. (Currently Amended) A method, in a data processing system, for resource allocation of a plurality of tasks carrying penalties based on their completion time, the method comprising:

assigning the plurality tasks to one or more resources;

assigning start times for the plurality of tasks such that expected penalties for completion times of the plurality of tasks are minimized, wherein the expected penalties are minimized by ~~continually~~ repeatedly assigning tasks and reevaluating start times for the plurality of tasks based on a plurality of predictable potential next events;

allocating ~~thinking~~ think time for each problem instance variant of the plurality of predictable potential next events into separate ~~thinking~~ think time partitions within each time slot for determining a best solution for each problem instance variant of [[a]] the plurality of the predictable potential next events, wherein an ~~optimal~~ amount of think time is calculated for each problem instance variant of the plurality of the predictable potential next events, wherein each time slot is divided into a plurality of separate think time partitions, wherein allocating think time for each problem instance variant of the plurality of predictable potential next events includes:

reserving a first amount of time for performing an initial algorithm;

allocating a second amount of time for performing a randomized algorithm, wherein the randomized algorithm is a next algorithm;

during each allocated ~~thinking~~ think time partition, allocating resources for a problem instance variant of a predicted next event at a predicted time at which the predicted next event may occur, wherein allocating the resources for the problem instance variant of the predicted next event includes:

executing the initial algorithm to form a preliminary solution;

recording a seed value of zero to indicate that a current solution is the preliminary solution; and

repeatedly executing the randomized algorithm until an event occurs or the second amount of time expires;

responsive to the randomized algorithm forming a solution that is better than a previous solution, updating the seed value; and

assigning resources for queued tasks based upon an actual next event and an actual time of occurrence, wherein assigning the resources for the queued tasks includes:

executing an algorithm that produced the best solution and assigning the resources based on results of the algorithm, wherein the algorithm is one of the initial algorithm and the randomized algorithm.

2-6. (Canceled)

7. (Currently Amended) The method of claim [5] 1, wherein the step of assigning resources for queued tasks based upon an actual next event and an actual time of occurrence includes:

determining whether a best solution was found using the initial algorithm or the randomized algorithm; and

responsive to the best solution being found using the initial algorithm, executing the initial algorithm and assigning resources based on results of the initial algorithm.

8. (Original) The method of claim 7, further comprising:

responsive to the best solution being found using the randomized algorithm, executing the randomized algorithm using the seed value and assigning resources based on results of the randomized algorithm.

9. (Original) The method of claim 8, further comprising:

assigning only immediately starting tasks.

10. (Currently Amended) The method of claim [5] 1, wherein an event is one of a job arrival, a task completion, a data change arrival, a managerial schedule request, and a termination request.

11. (Original) The method of claim 10, wherein a job includes one or more tasks.

12. (Canceled)

13. (Currently Amended) An apparatus, in a data processing system, for resource allocation of a plurality of tasks carrying penalties based on their completion time, the apparatus comprising:

a processor running a scheduler; and

at least one module coupled to the scheduler, wherein the scheduler provides:

means for assigning the plurality tasks to one or more resources;

means for assigning start times for the plurality of tasks such that expected penalties for completion times of the plurality of tasks are minimized, wherein the expected penalties are minimized by ~~continually~~ repeatedly assigning tasks and reevaluating start times for the plurality of tasks based on a plurality of predictable potential next events;

means for allocating ~~thinking~~ think time for each problem instance variant of the plurality of predictable potential next events into separate ~~thinking~~ think time partitions within each time slot for determining a best solution for each problem instance variant of [[a]] the plurality of the predictable potential next events, wherein an ~~optimal~~ amount of think time is calculated for each problem instance variant of the plurality of the predictable potential next events, wherein each time slot is divided into a plurality of separate think time partitions, wherein the means for allocating think time for each problem instance variant of the plurality of predictable potential next events includes:[[;]]

means for reserving a first amount of time for performing an initial algorithm;

means for allocating a second amount of time for performing a randomized algorithm, wherein the randomized algorithm is a next algorithm;

means for allocating resources, during each allocated ~~thinking~~ think time partition, for a problem instance variant of a predicted next event at a predicted time at which the predicted next event may occur, wherein the means for allocating the resources for the problem instance variant of the predicted next event includes:

means for executing the initial algorithm to form a preliminary solution;

means for recording a seed value of zero to indicate that a current solution is the preliminary solution; and

means for repeatedly executing the randomized algorithm until an event occurs or the second amount of time expires;

responsive to the randomized algorithm forming a solution that is better than a previous solution, means for updating the seed value; and

means for assigning resources for queued tasks based upon an actual next event and an actual time of occurrence, wherein the means for assigning the resources for the queued tasks includes:

means for executing an algorithm that produced the best solution and assigning the resources based on results of the algorithm, wherein the algorithm is one of the initial algorithm and the randomized algorithm.

14. (Currently Amended) A computer program product comprising a computer readable medium having encoded thereon computer usable program code for use within a data processing system for resource allocation of a plurality of tasks carrying penalties based on their completion time, the computer program product comprising:

computer usable program code for assigning the plurality tasks to one or more resources; and

computer usable program code for assigning start times for the plurality of tasks such that expected penalties for completion times of the plurality of tasks are minimized, wherein the expected penalties are minimized by ~~continually~~ repeatedly assigning tasks and reevaluating start times for the plurality of tasks based on a plurality of predictable potential next events;

computer usable program code for allocating ~~thinking~~ think time for each problem instance variant of the plurality of predictable potential next events into separate ~~thinking~~ think time partitions within each time slot for determining a best solution for each problem instance variant of [[a]] the plurality of the predictable potential next events, wherein an ~~optimal~~ amount of think time is calculated for each problem instance variant of the plurality of the predictable potential next events, wherein each time slot is divided into a plurality of separate think time partitions, wherein the computer usable program code for allocating think time for each problem instance variant of the plurality of predictable potential next events includes:[[;]]

computer usable program code for reserving a first amount of time for performing an initial algorithm;

computer usable program code for allocating a second amount of time for performing a randomized algorithm, wherein the randomized algorithm is a next algorithm;

computer usable program code for allocating resources, during each allocated ~~thinking~~ think time partition, for a problem instance variant of a predicted next event at a predicted time at which the predicted next event may occur, wherein the computer usable program code for allocating the resources for the a problem instance variant of the predicted next event includes:

computer usable program code for executing the initial algorithm to form a preliminary solution;

computer usable program code for recording a seed value of zero to indicate that a current solution is the preliminary solution; and

computer usable program code for repeatedly executing the randomized algorithm until an event occurs or the second amount of time expires;

responsive to the randomized algorithm forming a solution that is better than a previous solution, computer usable program code for updating the seed value; and computer usable program code for assigning resources for queued tasks based upon an actual next event and an actual time of occurrence, wherein the computer usable program code for assigning the resources for the queued tasks includes:

computer usable program code for executing an algorithm that produced the best solution and assigning the resources based on results of the algorithm, wherein the algorithm is one of the initial algorithm and the randomized algorithm.

15-19. (Canceled)

20. (Currently Amended) The computer program product of claim [18] 14, wherein the instructions for assigning resources for queued tasks based upon an actual next event and an actual time of occurrence includes:

computer usable program code for determining whether a best solution was found using the initial algorithm or the randomized algorithm; and

computer usable program code for responsive to the best solution being found using the initial algorithm, executing the initial algorithm and assigning resources based on results of the initial algorithm.

21. (Previously Presented) The computer program product of claim 20, further comprising:

computer usable program code, responsive to the best solution being found using the randomized algorithm, for executing the randomized algorithm using the seed value and assigning resources based on results of the randomized algorithm.

22. (Previously Presented) The computer program product of claim 21, further comprising:

computer usable program code for assigning only immediately starting tasks.

23. (Currently Amended) The computer program product of claim [18] 14, wherein an event is one of a job arrival, a task completion, a data change arrival, a managerial schedule request, and a termination request.

24. (Original) The computer program product of claim 23, wherein a job includes one or more tasks.